IN THE CLAIMS

Claims 36 – 45 have been added. Claims 8 – 10, 12, 13, 16, 18, 21, 22, 24, 27, 34 and 35 have been cancelled. Claims 11, 19, and 25 have been amended.

Claims 1 – 10 (cancelled).

11. (currently amended) ~~The computing system of claim 8, wherein~~ A computing system for performing a decryption operation on an encrypted packet, comprising:

a network driver to regulate said decryption operation and to transmit a decryption command;

a host memory to store the encrypted packet;

a controller to receive the encrypted packet and to perform said decryption operation after receiving said decryption command from the network driver;

a network interface to specify an interrupt handler latency value to the controller, said interrupt handler latency value [[is]] being based on a specific number of bytes being decrypted in the controller;

a bus providing electronic communication among said network driver, said host memory and said controller, said controller asserting an interrupt and ~~the interrupt is asserted~~ after the specific number of bytes have been decrypted in the controller and before the decrypted packet is transferred back from the controller to the host memory.

Claims 12 – 18 (cancelled).

19. (currently amended) ~~The method of claim 16, wherein said~~ A method of decrypting an encrypted packet received by a computing system, comprising:

receiving said encrypted packet from a network and transferring said encrypted

packet to a host memory;

issuing a decryption command to a controller;

specifying an interrupt handler latency value to the controller, the interrupt handler latency value [[is]] being based on a specific number of bytes being decrypted in the controller;

transferring said encrypted packet to said controller;

converting said encrypted packet to a decrypted packet; and

transferring said decrypted packet to the host memory, wherein [[the]] an interrupt is asserted after the specific number of bytes have been decrypted in the controller and before the decrypted packet has been transferred from the controller to the host memory.

Claims 20 – 24 (cancelled).

25. (currently amended) ~~The device of claim 22, wherein said~~

A program code storage device, comprising:

a machine-readable storage medium; and

machine-readable program code, stored on the machine-readable storage medium, the machine-readable program code having instructions that when executed cause a computer system to:

receive an encrypted packet from a network and transfer said encrypted packet to a host memory;

issue a decryption command to a controller;

specify an interrupt handler latency value to the controller, the interrupt handler latency value [[is]] being based on a specific number of bytes being decrypted in the

controller;

transfer said encrypted packet to said controller;

convert said encrypted packet to a decrypted packet; and

transfer said decrypted packet to the host memory, wherein [[and the]] an interrupt is asserted after the specific number of bytes have been decrypted in the controller and before the decrypted packet has been transferred from the controller to the host memory.

Claim 26 - 35. (cancelled)

36. (new) A method of decrypting an encrypted packet received by a network interface in a computing system, comprising:

receiving said encrypted packet from a network and transferring said encrypted packet to a host memory;

issuing a decryption command to a controller;

specifying an interrupt handler latency value to the controller, the interrupt handler latency value being based on a specific number of bytes being decrypted in the controller; and

transferring said encrypted packet to said controller which converts the encrypted packet to a decrypted packet and transfers the decrypted packet to the host memory, wherein an interrupt is asserted after a specific number of bytes have been decrypted in the controller and before the decrypted packets have been transferred from the controller to the host memory.

37. (new) The method of claim 36, wherein the encrypted packet is transferred to the host memory via direct memory access (DMA).

38. (new) The method of claim 36, further including a network driver to parse the encrypted packet at the network interface, match the encrypted packet with a corresponding security association, and to instruct that the corresponding security association is transferred to the controller with the encrypted packet.

39. (new) A program code storage device, comprising:

a machine-readable storage medium; and

machine-readable program code, stored on the machine-readable storage medium, the machine-readable program code having instructions that when executed cause a network interface to:

receive an encrypted packet from a network and transfer said encrypted packet to a host memory;

issue a decryption command to a controller;

specify an interrupt handler latency value to the controller, the interrupt handler latency value being based on a specific number of bytes being decrypted in the controller; and

transfer said encrypted packet to said controller which converts the encrypted packet to a decrypted packet and transfers the decrypted packet to the host memory, wherein an interrupt is asserted after a specific number of bytes have been decrypted in the controller and before the decrypted packets have been transferred from the controller to the host memory.

40. (new) The device of claim 39, wherein the encrypted packet is transferred to the host memory via direct memory access (DMA).

41. (new) The device of claim 39, further including a network driver to parse the

encrypted packet at the network interface, match the encrypted packet with a corresponding security association, and to instruct that the corresponding security association is transferred to the controller with the encrypted packet.

42. (new) The method of claim 19, wherein the encrypted packet is transferred to the host memory via direct memory access (DMA).

43. (new) The method of claim 19, further including a network driver to parse the encrypted packet at the network interface, match the encrypted packet with a corresponding security association, and to instruct that the corresponding security association is transferred to the controller with the encrypted packet.

44. (new) The device of claim 25, wherein the encrypted packet is transferred to the host memory via direct memory access (DMA).

45. (new) The device of claim 25, further including a network driver to parse the encrypted packet at the network interface, match the encrypted packet with a corresponding security association, and to instruct that the corresponding security association is transferred to the controller with the encrypted packet.